

FANS 13.4 – Miten re- faktoroin SAS9 sisältöä SAS Viyan muottiin

13.4.2023
Jarno Lindqvist



Agenda

9:00-10:00

- Miksi re-faktoroida, kun muutenkin toimii?
- Näin konvertoit EG projektin SAS Studion formaattiin
- Vilkaistu DIS-migraatioon (jos aikaa)
- Yhteenveto

Yleistä SAS Viyasta

- SAS Viya on muistinvaraiseen CAS-laskentaan kykenevä alusta joka on rakennettu pilvinatiiviksi.
- SAS Studio on ensimmäinen paikka hallita SAS9:sta tuotua sisältöä, sillä sinne voi tuoda niin koodin, EG projektit sekä DIS jobit
- SAS Viya on taaksepäin yhteensopiva SAS9 koodin suhteen ja niitä voidaan ajaa Viyan Compute Serverillä
- Muistinvaraisen CAS-laskennan hyödyntäminen vaatii joitakin SAS koodiin tehtäviä muutoksia. Muutostarpeen arviointia varten on tehty **Content Assessment**

Content Assessment tutkii SAS9 koodin ja projektit ja tuottaa raportin niiden Viya-yhteensopivuudesta

SAS 9 System Evaluation

Analyzes SAS 9.4 system and configuration details to identify known deployment issues.



SAS 9 Inventory

Counts and collects data about what exists on the SAS 9 system.



SAS 9 Publish

Prepares results for reporting.



SAS 9 Profile

Provides details about selected SAS artifacts.



SAS 9 Code Check

Validates that code will migrate to SAS Viya.



SAS 9 Gather SAS Code


Collects and organizes SAS Code from artifacts for Code Check.
Requires SAS 9 Profile



Mitä tarkoittaa refaktorointi SASilla

- Hyhmäinen termi, mutta yleisesti tarkoittaa SAS9 koodin, projektien ja jobien muokkaamista SAS Viya, CAS ja cloud –yhteensopivaksi
- Toki 1:1 siirtokin toimii, mutta silloin ei ehkä saada kaikkia hyötyjä irti ja ylläpito voi olla työläämpää, esim. lukuisten koodi-stepien takia
- Motivaattoreita:
 - Ylläpidettävämpi koodi ja projektit
 - Viyan uusia ominaisuuksia saadaan hyötykäyttöön
 - CAS:illa parempi suorituskyky raskaaseen laskentaan

Aiheesta useita SAS-blogeja

 | BLOGS All SAS ▾ 🔍 🌐 📄 🗃️ 👤

How to refactor SAS code to leverage SAS Viya 🗨️ 5


By [Steven Sober](#) on [SAS Users](#) | October 2, 2020 Topics | [Programming Tips](#)

If you're a SAS programmer who now uses SAS Viya and CAS, it's worth your time to optimize your existing programs to take advantage of the new environment. This post is a continuation of my SAS Global Forum 2020 paper [Best Practices for Converting SAS® Code to Leverage SAS® Cloud Analytic Services](#) and my [SGF 2020 Super Demo](#).

The best approach for refactoring SAS code for SAS Viya has a few steps:

- First, "lift and shift" your existing code to run successfully in the compute server for SAS Viya.
- Next, create CASLIB statements to all of your data sources: i.e. sas7bdat, CSV files, parquet files, relational databases, cloud data sources, etc.
- Finally, identify the longest running steps so you know where you have the biggest opportunities. For example, look at steps where the "real time" is 30 minutes or longer, as well as steps that are CPU bound. CPU-bound steps are steps where the CPU time is equal to or greater than the real time for that step.

To help us identify those steps we can leverage a new utility to analyze SAS logs and create reports to help us understand the Real Time and CPU Time for each step. Read on to learn more about this final step in the code refactoring process.



<https://blogs.sas.com/content/sgf/2020/10/02/refactor-sas-code-viya/>

Refaktorointi sidoksissa vahvasti SAS Studioon

SAS Enterprise Guide



SAS Data Integration Studio



Base SAS ja SAS-koodi



SAS Studio



Koodaa

SAS-koodieditori,
jossa ennakoiva
tekstinsyöttö ja
pikalinkit help-
sivuihin

Rakenna

SAS-koodi, valmiit
taskit, Flow stepit,
kyselyt

Jaa

GIT-integraatio,
jaetut custom
stepit ja taskit

SAS Viya 2023.01 version vakio-stepit

SAS STUDIO BASIC

SAS Steps : Shared

Data (Input and Output)

- Export
- File
- Import
- Table

Develop

- Python Program
- SAS Program

Transform Data

- Branch Rows
- Calculate Columns
- Filter Rows
- Insert Rows
- Manage Columns
- Query
- Remove Duplicates
- Sort

Integrate

- Execute Decisions
- Implement SCD
- Load Table
- Merge Table

Enrichment

- Geocode Data
- Verify & Geocode Addresses - Loqate
- Verify Email Addresses - Loqate

SAS STUDIO ANALYST

SAS Steps : Shared

Data (Input and Output)

- Export
- File
- Import
- Table

Develop

- Python Program
- SAS Program

Transform Data

- Branch Rows
- Calculate Columns
- Filter Rows
- Insert Rows
- Manage Columns
- Query
- Remove Duplicates
- Sort

Integrate

- Execute Decisions
- Implement SCD
- Load Table
- Merge Table

Enrichment

- Geocode Data
- Verify & Geocode Addresses - Loqate
- Verify Email Addresses - Loqate

SAS STUDIO ENGINEER

SAS Steps : Shared

Data (Input and Output)

- Export
- File
- Import
- Table

Develop

- Python Program
- SAS Program

Transform Data

- Branch Rows
- Calculate Columns
- Filter Rows
- Insert Rows
- Manage Columns
- Query
- Remove Duplicates
- Sort

Integrate

- Execute Decisions
- Implement SCD
- Load Table
- Merge Table

Enrichment

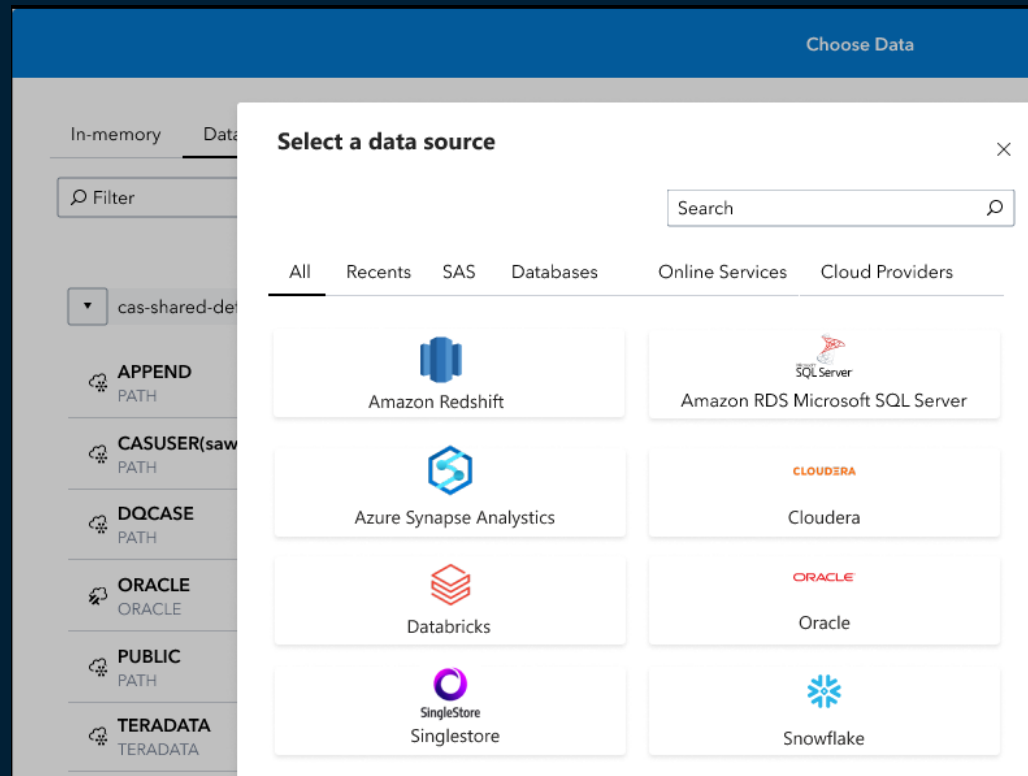
- Geocode Data
- Verify & Geocode Addresses - Loqate
- Verify Email Addresses - Loqate

Stepejä tulossa paljon lisää tänä vuonna

Roadmap Ahead 2023

Datalähteiden keskitetty hallinta

Sekä CAS että Compute kirjastoja hallitaan jatkossa yhdestä samasta paikasta helposti ja intuitiivisesti



SAS Studion ominaisuudet kehittyvät

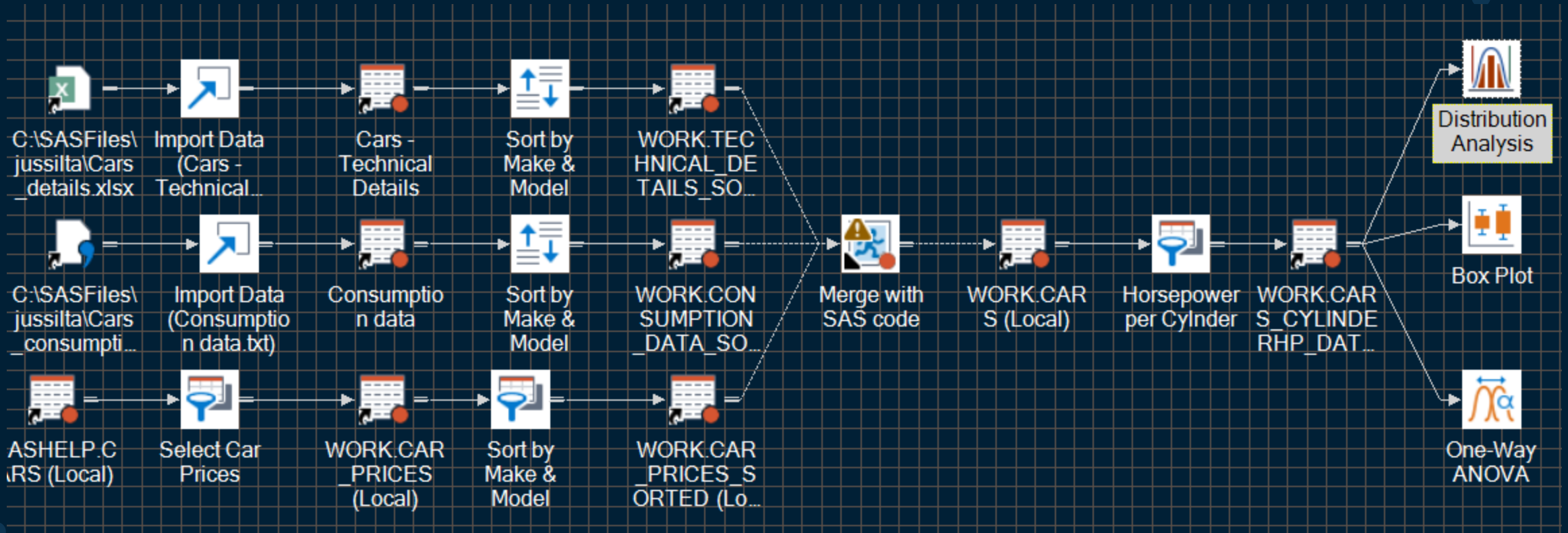
Tämän vuoden aikana SAS Studio saa paljon uusia stepejä ja yhteensopivuus EG:n ja DI Studion kanssa paranee

- Analyst step - One-Way Frequencies
- Analyst step - Transpose
- Analyst step - Rank
- Analyst step - Bar Chart
- Analyst step - Bar-Line Chart
- Analyst step - Union step
- Analyst step - Pie Chart
- Analyst step - Mosaic plot
- Analyst step - Model Registration
- Engineer step - Flow Control - Loop
- Analyst step - Model Scoring
- Analyst step - Look Up

SAS EG -migraatio

- SAS 9:n EG projekteista tulee SAS Viyassa SAS Studio floweja
- Kaikki EG taskit migroituvat vähintään SAS Studion koodi-stepeiksi
- EG:n **query taskeista** tulee natiiveja SAS Studio query stepejä, sis. select, join, filter ja sort, calculated columns ja expressionit.
- EG:n **import taskista** tulee SAS Studion import step, tukee .XLS, CSV ja fixed length tiedostoja.
- EG:n **export taskista** tulee SAS Studion export step.
- SAS Viyan Environment Manager tekee migration jälkeen raportin josta näkee miten kattavasti import meni läpi, sekä mahdolliset varoitukset

Alkuperäinen EG-projekti



Import vaihe

Import

Import Mapping Sets

Source
Mapping
Preview
Results

Preview

Package to import: Cars_project.egp

✔ If the summary information below is correct, select Import to begin the import process.

1
Content Items

CONTENT ITEMS

1 SAS Enterprise Guide project

Results

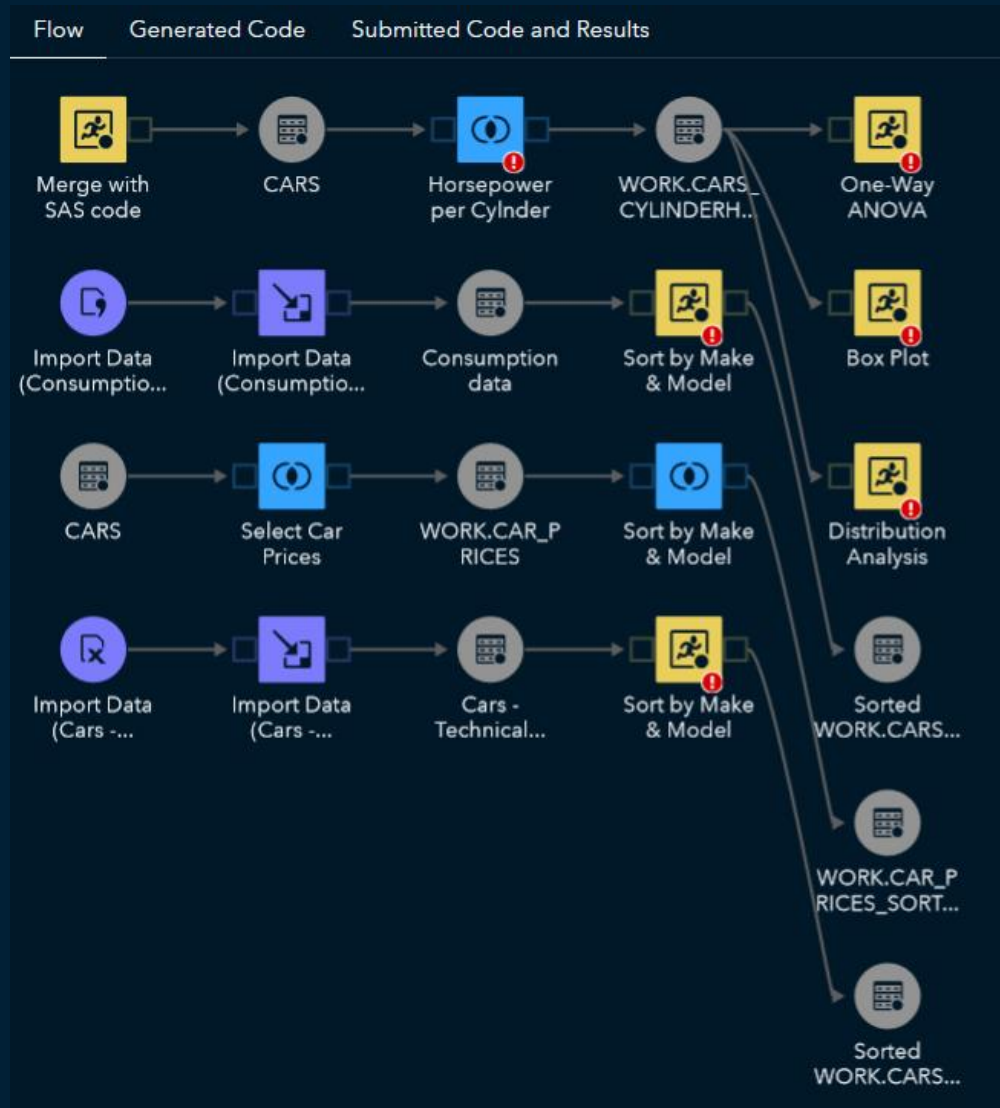
✔ Import job has completed

Total time to import Cars_project.egp: 0:00:05

Show column abbreviations

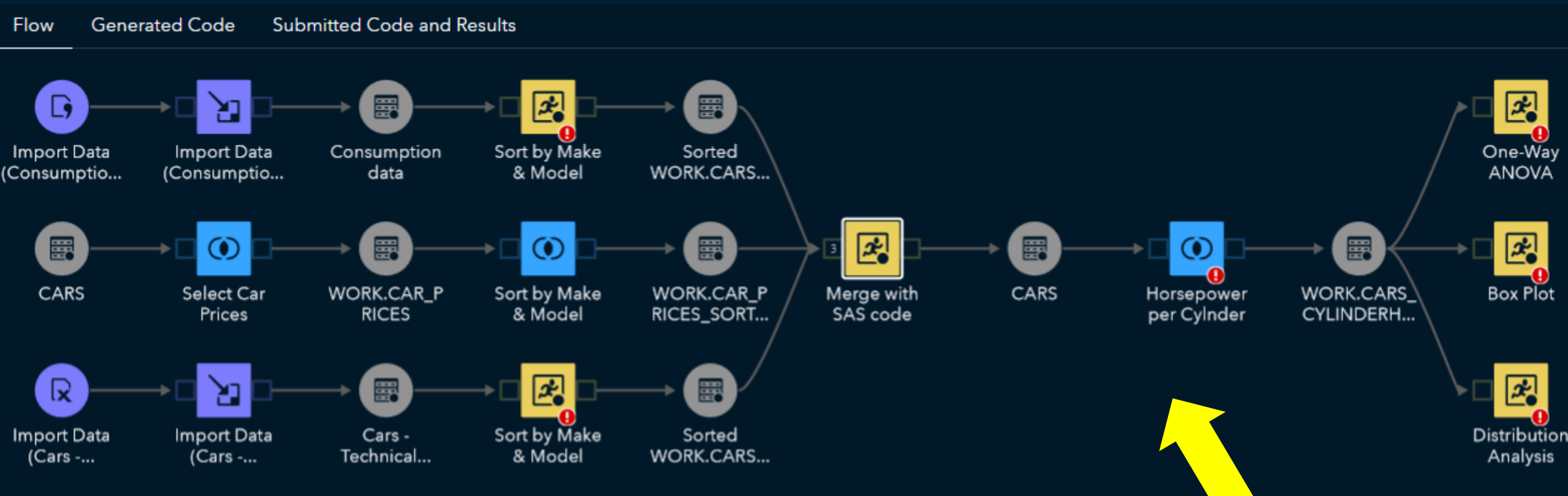
Object Type		✔ Complet...	⚠ Warnings	✖ Errors
🔄 SAS Enterprise Guide project	(1)	0	1	0

Import-vaiheen jälkeen SAS Studiossa

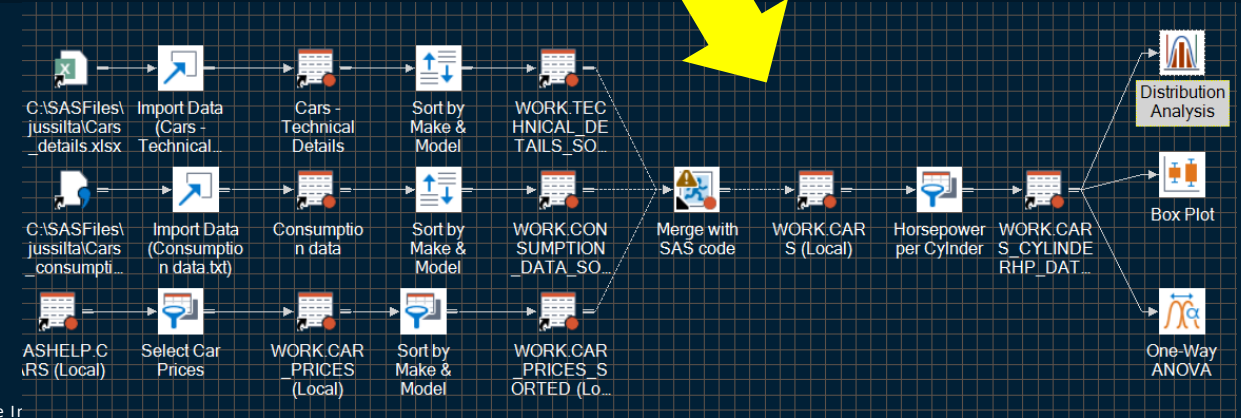


Näyttää aluksi oudolta mutta tarkistelun jälkeen tuo Merge koodinpätkä vain on kadottanut kiinnityksen edellisiin tauluihin...

Langat uudestaan kiinni



Kun puuttuvat langat yhdistellään takaisin kiinni ja uudelleenjärjestellään, niin näyttää jo samalta projektilta!



Viittaukset lähdetiedostoihin kuntoon

Import Data (Consumption data.txt)




Options Node Notes

▼ Data

Place:

SAS Content

File location:

/Users/Jarno.Lindqvist@sas.com/My Folder/FANS Apr 202 ...   

File name:

Cars_consumption.csv

File type:

Comma delimited (*.csv) ▼

File type:

Comma delimited (*.csv) ▼

Encoding:

UTF-8 ▼

Analysis Options Update Options

Guessing rows: ?

200

Delimiter:

Comma ▼

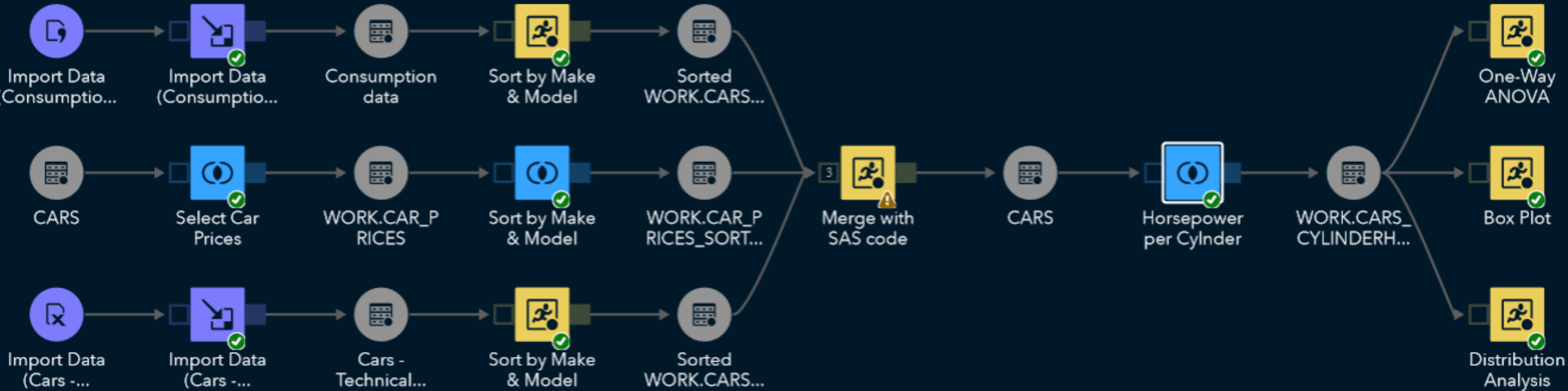
Column names are in first row of input file

Rename column names to comply with SAS naming conventions

Viittaukset kuntoon ja "Guessing rows" kannattaa tarkistaa, että kentistä varmasti tulee oikean mittaiset

Näiden jälkeen projektin voi ajaa

Flow Generated Code Submitted Code and Results



Saadaan vihreitä lippuja kaikkiin projektin steppeihin – paitsi Merge antaa varoituksen eripituisista kentistä, mutta se on normaalia, kun yhdistellään eri lähteistä

Palastellaan swim laneihin osiot

Flow Generated Code Submitted Code and Results

^ Prep data

Import Data (Consumptio...)

Import Data (Consumptio...)

Consumption data

Sort by Make & Model

Sorted WORK.CARS...

CARS

Select Car Prices

WORK.CAR_PRICES

Sort by Make & Model

WORK.CAR_PRICES_SORT...

Merge with SAS code

CARS

Import Data (Cars -...)

Import Data (Cars -...)

Cars - Technical...

Sort by Make & Model

Sorted WORK.CARS...

^ Analyze data

CARS

Horsepower per Cylinder

WORK.CARS_CYLINDERH...

One-Way ANOVA

Box Plot

Distribution Analysis

Submission Order

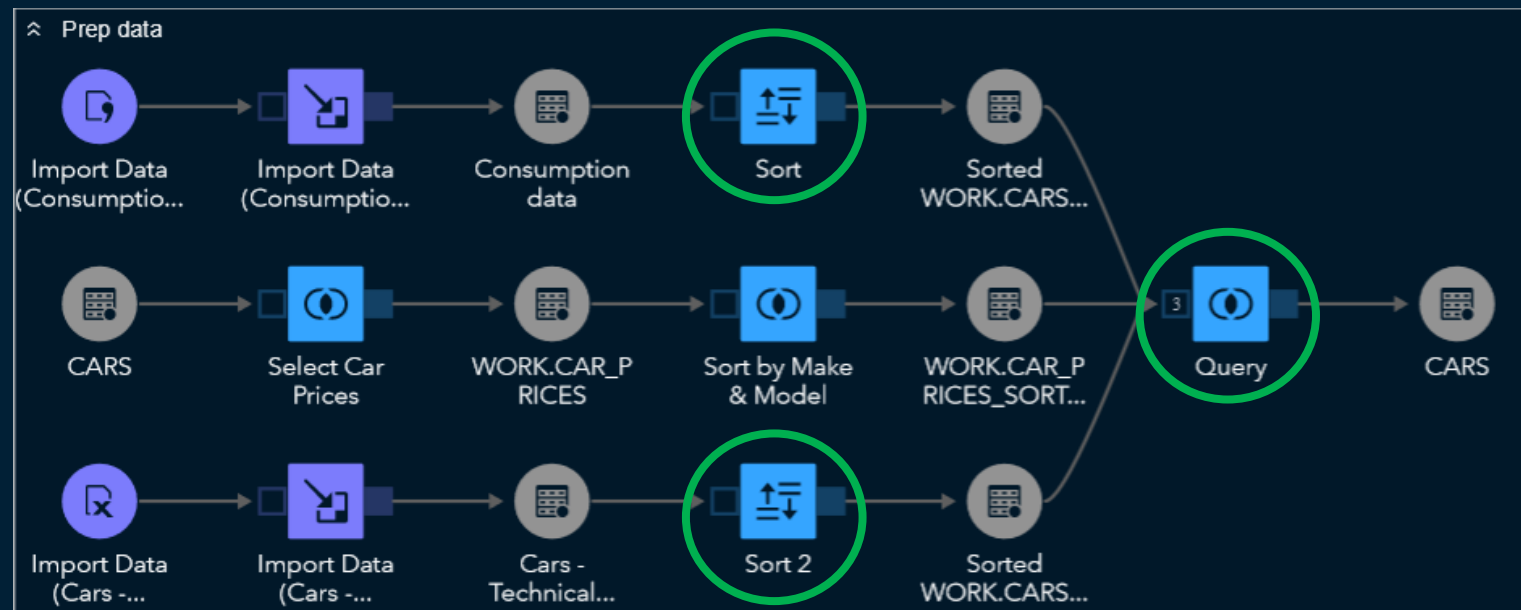
You can specify the order in which swimlanes are run. ⓘ

Enable submission order

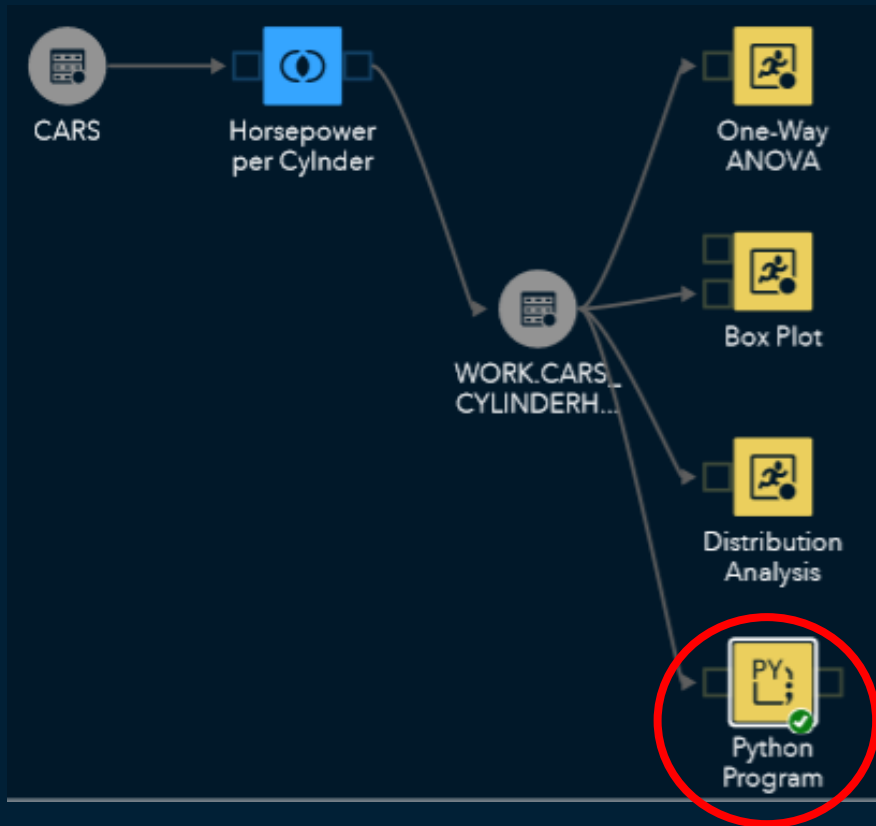
<input type="checkbox"/>	Order	Swimlane Name
<input type="checkbox"/>	1	Prep data
<input type="checkbox"/>	2	Analyze data

sas

Halutessaan voi korvata koodit SAS Studio stepeillä



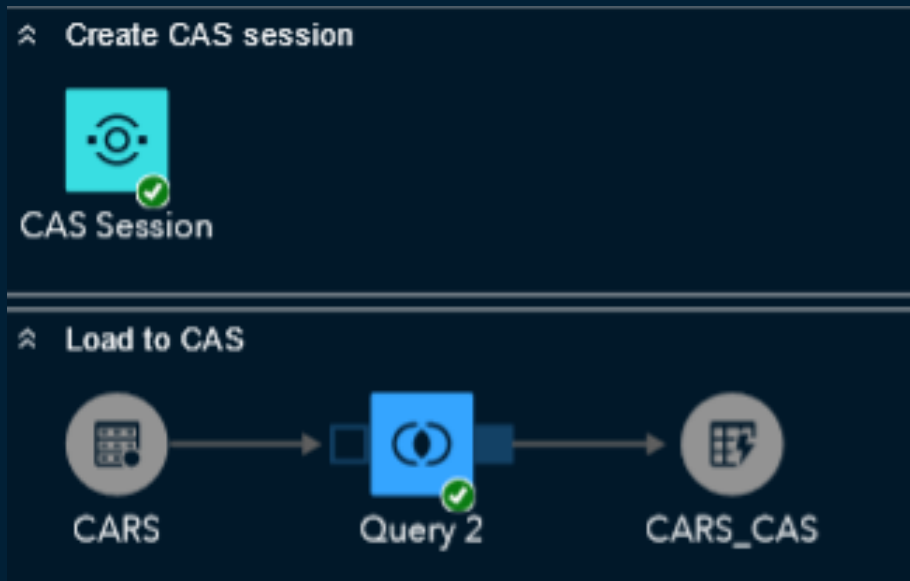
Tarvittaessa voidaan ottaa myös Pythonin voimat käyttöön



Code	Node	Notes
1		# Import Pandas
2		import pandas as pd
3		
4		# Import SAS dataset to dataframe
5		data = SAS.sd2df("work.CARS_CYLINDERHP_DATA")
6		
7		# Frequencies in Actual Order
8		datax = data['Make'].value_counts().sort_index()
9		
10		# Create a dataframe
11		datax = pd.DataFrame({
12		'Make': datax.index,
13		'Frequency': datax.values,
14		'Percent': ((datax.values/datax.values.sum()*100).round(2),
15		'Cumulative Frequency': datax.values.cumsum(),
16		'Cumulative Percent': ((datax.values.cumsum()/datax.values.sum()*100)\
17		.round(2)
18		})
19		print(datax.head(100))

	Make	Frequency	Percent	Cumulative Frequency	Cumulative Percent
0	Acura	7	1.87	7	1.87
1	Audi	19	5.08	26	6.95
2	BMW	20	5.35	46	12.30
3	Buick	9	2.41	55	14.71
4	Cadillac	8	2.14	63	16.84
5	Chevrolet	27	7.22	90	24.06
6	Chrysler	15	4.01	105	28.07
7	Dodge	13	3.48	118	31.55
8	Ford	23	6.15	141	37.70
9	GMC	8	2.14	149	39.84
10	Honda	17	4.55	166	44.39

Jatkoanalyysi halutaan tehdä VA:n puolella niin ladataan vielä CASin muistiin



The screenshot shows the "Options" tab for the "CARS_CAS" table. The interface includes tabs for "Table Properties", "Options", "Published Columns", "Preview Data", "Node", and "Notes". Below the tabs, there is a descriptive text: "Specify the CAS output table options. By default, session-based CAS tables are generated." There are four options listed with checkboxes:

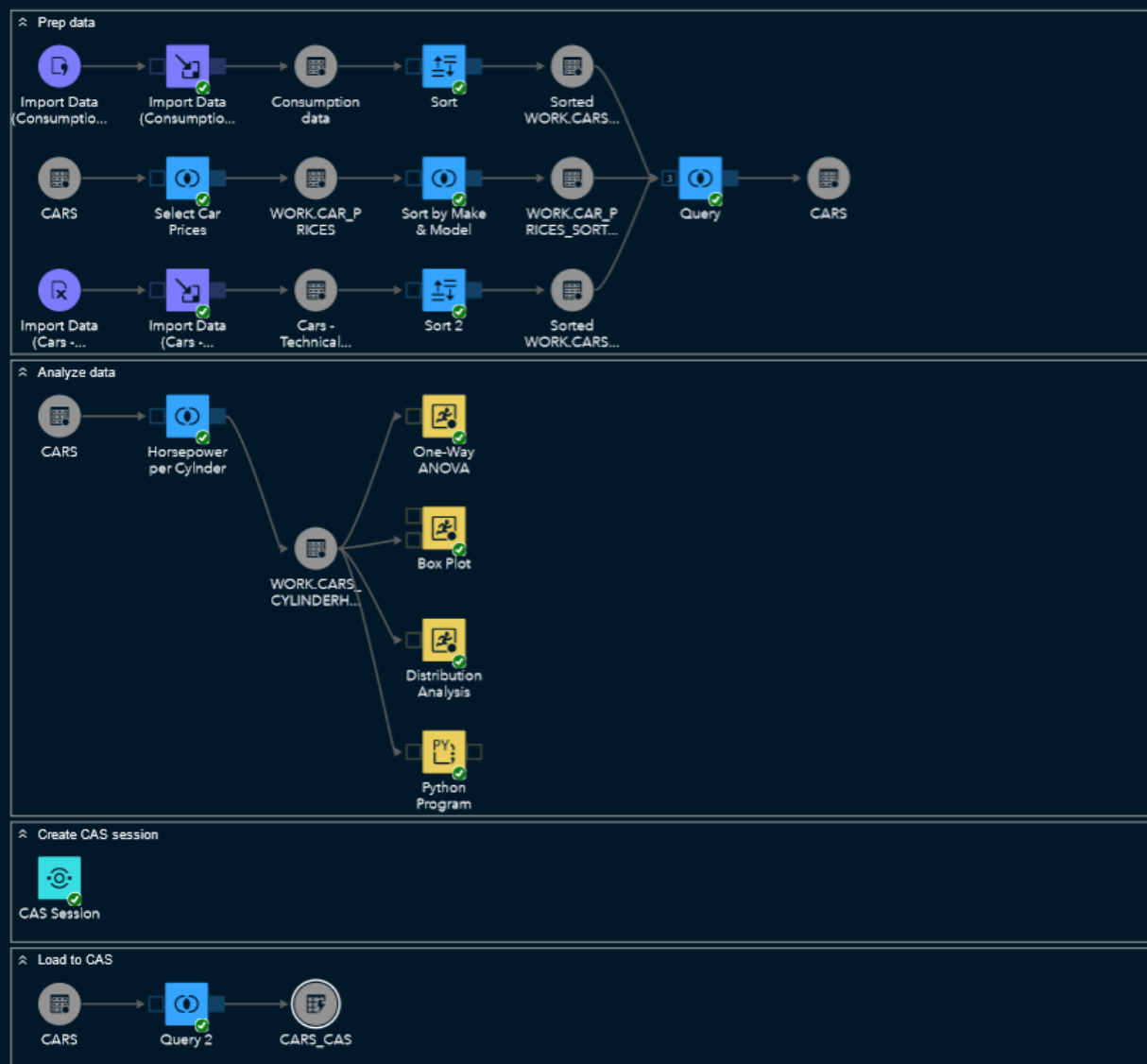
- Promote to a global in-memory table
- Drop and replace the table if it exists
- Save to persist the table on disk
- Replace the existing table saved to disk

Muodostetaan ensin CAS-sessio käyttämällä "CAS Session" custom stepiä.



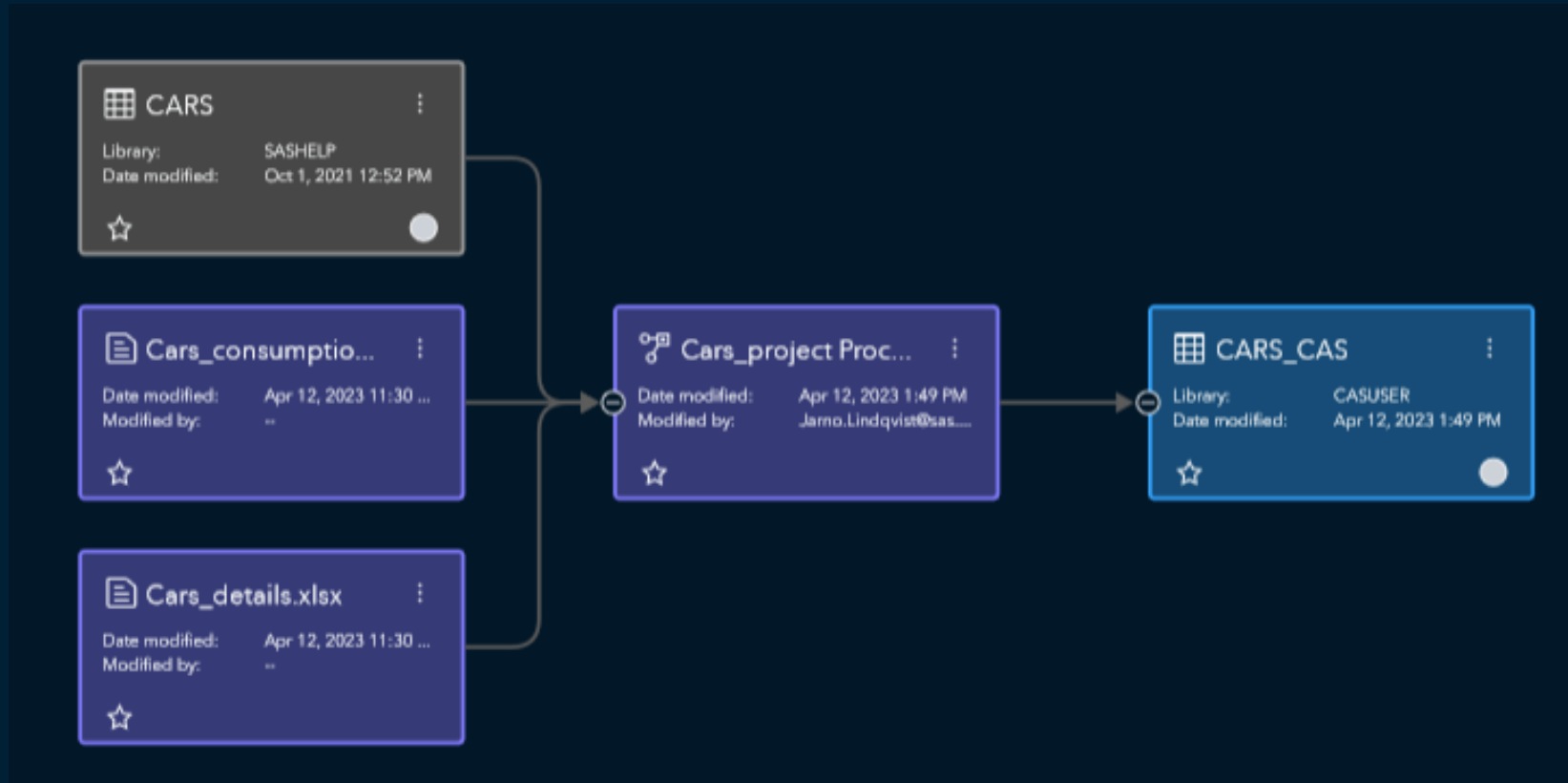
CASiin lataukseen voidaan käyttää tavallista Query Builder –nodea, kunhan muistaa laittaa "Promote to Global in-memory table" option päälle, niin se näkyy muissakin sessioissa kuten VA:ssa

Valmis refaktoroitu EG-jobi SAS Studiassa



- Datan lataus ulkoisista tiedostoista sekä yhdistely ja sorttaus
- SAS-analyysit kuten ennenkin ja lisäksi nyt myös Pythonilla tehty frekvenssianalyysi
- CAS session muodostus
- Nostetaan valmis SAS-taulu CASin muistiin

Sama jobi Lineagen kautta tarkasteltuna

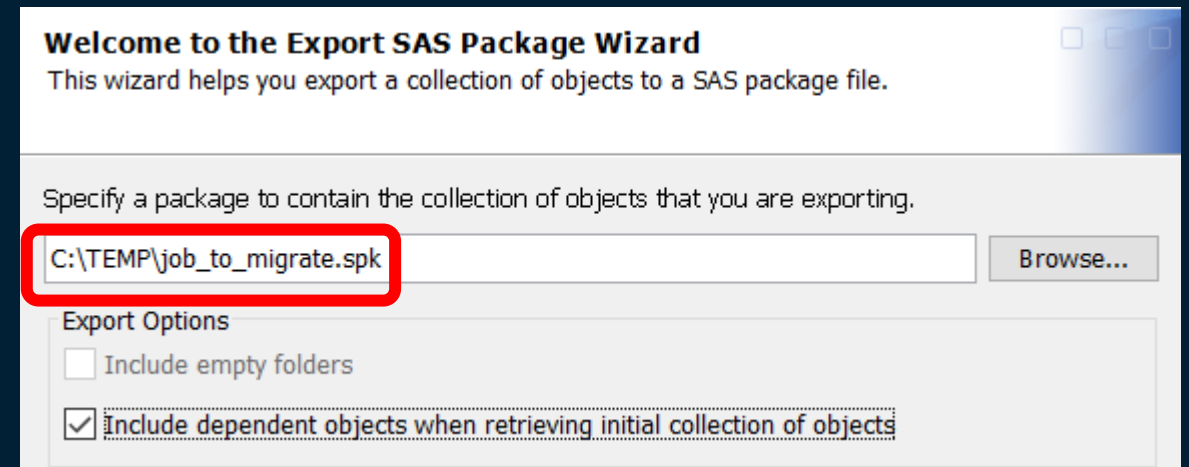
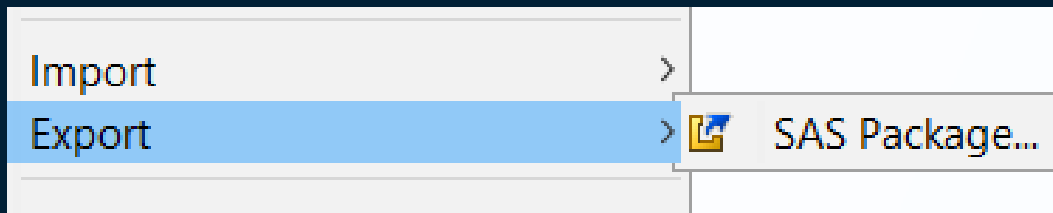
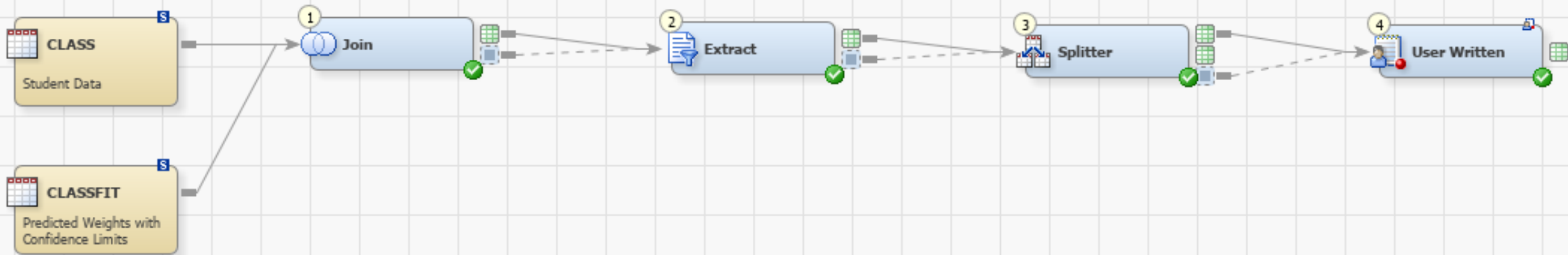


Lähde ja kohdedata näkyy Lineageessa, samoin juuri rakentamamme SAS Studio Flow. Lineageen on myöhemmin tulossa myös mahdollisuus tarkastella flown sisältöä.

SAS DI Studio -migraatio

- SAS 9:n DI Studio jobeista tulee SAS Viyassa SAS Studio floweja
- Toistaiseksi **ei ole** tukea kaikkien DIS transformaatioiden migraatiolle
- Nyt 1-to-1 tuettuja esimerkiksi Table Loader, Merge, Append
- Lopuista tulee koodi-stepejä SAS Studiossa
- 1-to-1 migration kehitysjärjestys noudattaa SAS9 Content Assessementeissa saatuja tuloksia
- SAS Viyan Environment Manager tekee migration jälkeen raportin josta näkee miten kattavasti import meni läpi, sekä mahdolliset varoitukset
- Jobien bulkki-migraatio voidaan tehdä komentorivityökaluilla (*createSASpackages, importSASpackages*)

Esimerkki DIS-jobin migraatiosta



Import muutamalla vaiheella

Source

Import file: job_to_migrate.spk

Content

View: All resources

Select all

- CLASS
- CLASSFIT
- DIS_job_to_migrate



Mapping

Content Users and Groups Tables Data R

Target location: /Public/Imported DIS jobs

Imported DIS jobs

- DIS_job_to_migrate.flw



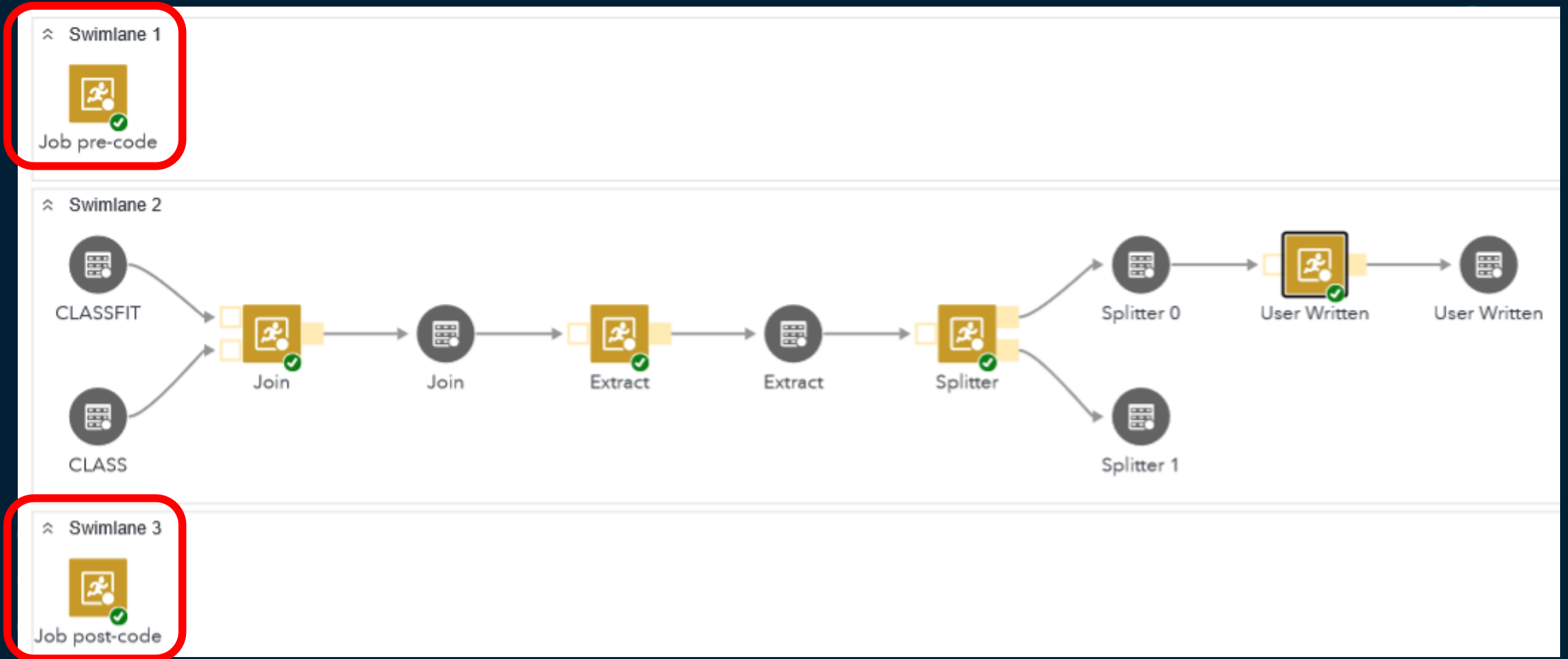
Object Type	Completed	Warnings	Errors	Failed	Skipped	Stopped
Job (1)	1	0	0	0	0	0

Import Details

Type: All

Name	State	Type	Path	Message
DIS_job_to_migrate	✓	Job	/Public/Imported DIS jobs/DIS_job_to_migrate.flw	The resource was successfully imported. "nodeName" "DIS_job_to_migrate" "Type" "ENTIRE PROJECT" "N..

Migroitu DIS-jobi SAS Studiassa



Paljonko refaktorointia kannattaa tehdä?

- Minimissään sen verran että toimii vastaavasti kuin SAS9:llä
- Koodi suoraviivaisinta tuoda, yleensä riittää dataviittausten muokkaus kohdalleen
- Muista merkkikonversio tarvittaessa, CAS on aina UTF8 ja normaalisti myös Compute
 - Helpottaa elämää kun tekee konversion ennen CASiin lataamista
- On makuasia, haluaako muokata koodina importoituneet asiat SAS Studio stepeiksi, mutta se saattaa helpottaa ylläpitoa jatkossa
- Jatkossa yhä useampi step tulee tukemaan 1:1 migraatiota joka osaltaan helpottaa prosessia – kysy SASilta jos kaipaat jotakin tiettyä toimintoa



sas viya

SAS just reinvented analytics. Again.

sas.com

